AD-A077 360 AIR FORCE GEOPHYSICS LAB HANSCOM AFB MA F/G 7/4
COMMUTE - A COMPUTER CODE FOR NONCOMMUTATIVE ALGEBRA. (U)
JUL 79 L S ROTHMAN
UNCLASSIFIED AFGL-TR-79-0152 NL

END
DATE FILMED
8-79
DDC

AFGL-TR-79-0152
ENVIRONMENTAL RESEARCH PAPERS, NO. 686

LEVEL II

COMMUTE-A Computer Code for Noncommutative Algebra

LAURENCE S. ROTHMAN

5 July 1979

OPTICAL PHYSICS DIVISION PROJECT 2310

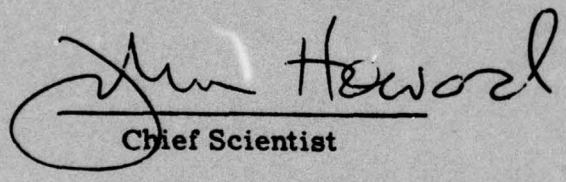
AIR FORCE GEOPHYSICS LABORATORY
HANSCOM AFB, MASSACHUSETTS 01731

AIR FORCE SYSTEMS COMMAND, USAF

This report has been reviewed by the ESD Information Office (OI) and is releasable to the National Technical Information Service (NTIS).

This technical report has been reviewed and
is approved for publication.

FOR THE COMMANDER



Chief Scientist

Qualified requestors may obtain additional copies from the Defense Documentation Center. All others should apply to the National Technical Information Service.

DISCLAIMER NOTICE

THIS DOCUMENT IS BEST QUALITY PRACTICABLE. THE COPY FURNISHED TO DDC CONTAINED A SIGNIFICANT NUMBER OF PAGES WHICH DO NOT REPRODUCE LEGIBLY.

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE	READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER: AFGL-TR-79-0152 AFGL-ERP-666	2. GOVT ACCESSION NO. / 3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle): COMMUTE – A COMPUTER CODE FOR NONCOMMUTATIVE ALGEBRA,	5. TYPE OF REPORT & PERIOD COVERED: Scientific. Interim
	6. PERFORMING ORG. REPORT NUMBER: ERP No. 666
7. AUTHOR(s): Laurence S. Rothman	8. CONTRACT OR GRANT NUMBER(s): Environmental research papers
9. PERFORMING ORGANIZATION NAME AND ADDRESS: Air Force Geophysics Laboratory (OP), Hanscom AFB, Massachusetts 01731	10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS: 61102F, 2310G103
11. CONTROLLING OFFICE NAME AND ADDRESS: Air Force Geophysics Laboratory (OP), Hanscom AFB, Massachusetts 01731	12. REPORT DATE: 5 July 1979 13. NUMBER OF PAGES: 30
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office)	15. SECURITY CLASS. (of this report): Unclassified 15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Atmospheric absorption
Molecular spectroscopy
Molecular structure
Infrared

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

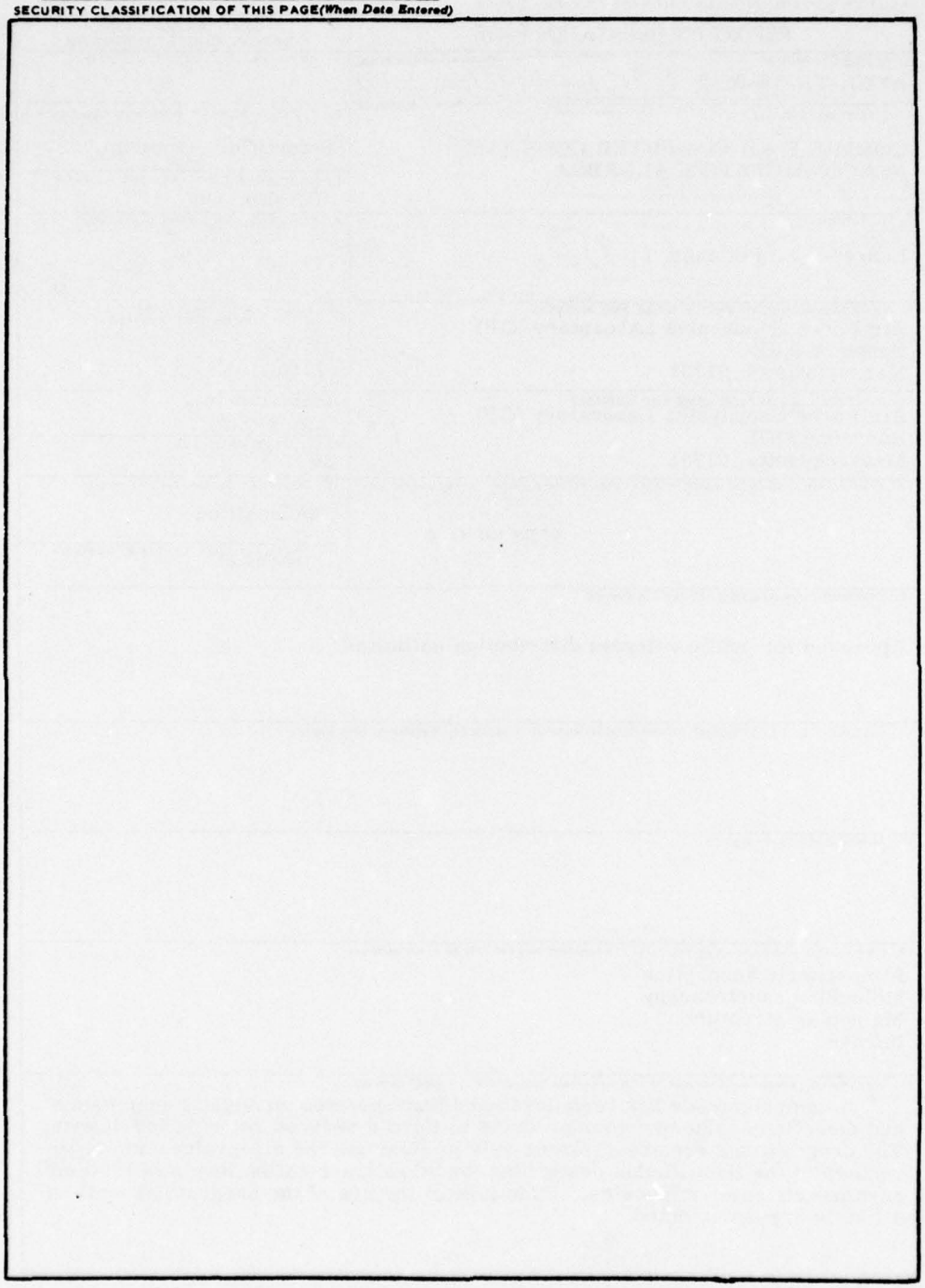
A computer code has been developed that operates on angular momentum and direction cosine operator products to form a reduced set of lower degree. The program has been used extensively in obtaining the eigenvalues and eigenvectors of the Hamiltonian describing the vibration-rotation levels of triatomic asymmetric rotor molecules. Examples of the use of the program as well as a listing are documented.

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

409578



Preface

I wish to acknowledge the many contributions and developmental efforts of S. A. Clough to the problems represented in this report and to the helpful discussions we have had.

Contents

COMMUTE – A Computer Code for Noncommutative Algebra

1. INTRODUCTION

In the course of investigations of infrared absorbing frequencies and line strengths of triatomic molecules of atmospheric interest, the main theoretical approach has invariably involved the necessity of calculating complex commutators of quantum operators. In particular, when dealing with the vibration-rotation Hamiltonian, one encounters commutators and anticommutators of angular momentum operators and related direction cosine operators between molecular-fixed axes and space-fixed axes. This report presents a computerized method of generating simplified series of operators, a so-called reduced set, from given initial commutators, anticommutators, or general operator polynomials. This computer program, which implements noncommutative algebra, is called COMMUTE.

2. THEORY

2.1 General Formulation

The general problem is one in which there is a commutator consisting of products of some fundamental operators, whose basic commutation relation is

(Received for publication 3 July 1979)

given. Through successive application of this fundamental commutation relation, the original commutator can be simplified into a series of anticommutators. This can be expressed symbolically as

$$C_0[\underline{O}'_0, \underline{O}''_0] = C_1\{\underline{O}'_1, \underline{O}''_1\} + C_2\{\underline{O}'_2, \underline{O}''_2\} + \ldots + C_i\{\underline{O}'_i, \underline{O}''_i\} + \ldots \quad , \qquad (1)$$

where the $\underline{O}'_i$ and $\underline{O}''_i$ are, in general, products of operators, the C_i are either pure real or pure imaginary numbers, and the brackets and braces are the standard notation for commutators and anticommutators, that is,

$$[\underline{A}, \underline{B}] = \underline{A}\underline{B} - \underline{B}\underline{A}$$

and

$$\{\underline{A}, \underline{B}\} = \underline{A}\underline{B} + \underline{B}\underline{A} \quad . \qquad (2)$$

The series in Eq. (1) in practice terminates rapidly; the number of terms is proportional to the degree of the original operators $\underline{O}'_0$ and $\underline{O}''_0$. The degree, that is, the sum of the powers of fundamental operators in the operator polynomial on the right side of Eq. (1), is at least one less than the original operator polynomial $[\underline{O}'_0, \underline{O}''_0]$. In fact, if the original commutator is of degree n, then the series will contain terms of degree n-1, n-3, n-5, etc. (Of course, terms may be missing altogether if their coefficients, C_i, are identically zero.) Furthermore, it follows from the preservation of hermiticity (or antihermiticity) that if C_0 is an imaginary number, then the coefficients of the anticommutators are all pure real. Conversely, if C_0 is real, the coefficients on the right side of Eq. (1) are pure imaginary.

These remarks will be demonstrated in the examples that follow.

2.2 Diagonalizing the Hamiltonian

The problem discussed in this section has been presented by Rothman and Clough[1] and typifies the use of program COMMUTE. A contact transformation

$$e^{i\underline{S}'}\,\underline{H}\,e^{-i\underline{S}'} \qquad (3)$$

is applied to the Hamiltonian to yield a transformed Hamiltonian expanded in a Taylor's series

---

1. Rothman, L.S. and Clough, S.A. (1971) J. Chem. Phys. 55:504.

$$\underline{H}'(\lambda) = \underline{H} + \sum_{n=1}^{\infty} \frac{\lambda^n i^n}{n!} [\underline{S}', \ldots [\underline{S}', \underset{(n)}{[\underline{S}', \underline{H}]}] \ldots] \quad , \tag{4}$$

where λ is an expansion parameter and $\underline{S}'$ is an Hermitian operator ($i = \sqrt{-1}$). Utilizing the order of magnitude arguments given in Reference 1, one arrives at expressions for the "rotational Hamiltonian", Eqs. (31) of Reference 1. $\underline{S}'$ is chosen to remove certain off-diagonal terms in the Hamiltonian and, as such, is itself formed of angular momentum operators. Thus, expressions for the rotational Hamiltonian contain numerous commutators of angular momentum operators. In Appendix B. 1, two representative commutators arising from the contact transformation are given.

An alternative scheme for calculating the rotational commutators has been presented in the work of Amat et al.[2] In their approach, the contribution of a chosen operator term is given as a function of commutators of various directions. As such, their procedure is the inverse of the one discussed here.

2.3 Reducing Operations

The rotational Hamiltonian through fourth order of a planar molecule in the XZ plane is of the form

$$\underline{H}^{\dagger\dagger} = h_v + A\underline{P}_z^2 + B\underline{P}_x^2 + C\underline{P}_y^2 + D\{\underline{P}_x, \underline{P}_z\} + \sum_{\alpha\beta\gamma\delta} \rho'_{\alpha\beta\gamma\delta} \underline{P}_\alpha \underline{P}_\beta \underline{P}_\gamma \underline{P}_\delta$$

$$+ \sum_{\alpha\beta\gamma\delta\epsilon\eta} \Phi'_{\alpha\beta\gamma\delta\epsilon\eta} \underline{P}_\alpha \underline{P}_\beta \underline{P}_\gamma \underline{P}_\delta \underline{P}_\epsilon \underline{P}_\eta \quad , \tag{5}$$

where h_v represents the vibrational contribution to the Hamiltonian; A, B, C, and D are coefficients of the quadratic terms in the angular momentum; the $\rho'_{\alpha\beta\gamma\delta}$ are coefficients of the quartic terms in angular momentum; and the $\Phi'_{\alpha\beta\gamma\delta\epsilon\eta}$ are coefficients of the sextic terms in angular momentum. For the case of orthorhombic symmetry, there are 15 nonvanishing $\rho'_{\alpha\beta\gamma\delta}$ and 105 $\Phi'_{\alpha\beta\gamma\delta\epsilon\eta}$. A great amount of the simplification can be obtained in the Hamiltonian by application of the commutation relations of angular momentum to these coefficients. The

---

2. Amat, G., Goldsmith, M., and Nielsen, H.H. (1957) J. Chem. Phys. 27:838.

resulting reduced form of $\underline{H}^{\dagger\dagger}$ for asymmetric rotors of orthorhombic symmetry thus contains only 19 coefficients (Kneizys et al[3]):

$$\begin{aligned}\underline{H}^{\dagger\dagger} = {} & h_v + A\,\underline{P}_z^2 + B\,\underline{P}_x^2 + C\,\underline{P}_y^2 \\ & + T_1\,\underline{P}_x^4 + T_2\,\underline{P}_y^4 + T_3\,\underline{P}_z^4 \\ & + T_4\,\{\underline{P}_y^2, \underline{P}_z^2\} + T_5\,\{\underline{P}_x^2, \underline{P}_z^2\} + T_6\{\underline{P}_x^2, \underline{P}_y^2\} \\ & + \Phi_1\,\underline{P}_x^6 + \Phi_2\,\underline{P}_y^6 + \Phi_3\,\underline{P}_z^6 + \Phi_4\,\{\underline{P}_x^2, \underline{P}_y^4\} \\ & + \Phi_5\,\{\underline{P}_y^2, \underline{P}_x^4\} + \Phi_6\,\{\underline{P}_y^2, \underline{P}_z^4\} + \Phi_7\,\{\underline{P}_z^2, \underline{P}_y^4\} \\ & + \Phi_8\,\{\underline{P}_z^2, \underline{P}_x^4\} + \Phi_9\,\{\underline{P}_x^2, \underline{P}_z^4\} + \Phi_{10}\,(\underline{P}_x^2\underline{P}_y^2\underline{P}_z^2 + \underline{P}_z^2\underline{P}_y^2\underline{P}_x^2)\ . \end{aligned} \tag{6}$$

2.4 Minimal Set of Operators

It has been shown by Watson[4] that the coefficients of the Hamiltonian of Eq. (6) are not all linearly independent for an asymmetric-top molecule. This gives rise to highly correlated coefficients when an attempt is made to estimate the coefficients in conjunction with a least squares approach with the experimental data (Flaud and Camy-Peyret, [5] Rothman and Clough[6]). The indeterminancy can be further removed by successive unitary transformations on $\underline{H}^{\dagger\dagger}$ of the form

$$\underline{H}_w = e^{i\underline{\mathcal{S}}}\,\underline{H}^{\dagger\dagger}\,e^{-i\underline{\mathcal{S}}}\ . \tag{7}$$

Successive transformations of this type again lead to a series of commutators analogous to Eq. (4). The choice of elements s_{pqr}, where

$$\underline{\mathcal{S}} = \sum_{pqr} s_{pqr}\,(\underline{P}_x^p\underline{P}_y^q\underline{P}_z^r + \underline{P}_z^r\underline{P}_y^q\underline{P}_x^p)\ , \tag{8}$$

---

3. Kneizys, F.X., Freedman, J.N., and Clough, S.A. (1966) J. Chem. Phys. 44:2552.
4. Watson, J.K.G. (1967) J. Chem. Phys. 46:1935; (1968) J. Chem. Phys. 48:4517.
5. Flaud, J.-M. and Camy-Peyret, C. (1973) Mol. Phys. 26:811.
6. Rothman, L.S. and Clough, S.A. (1975) Determination of Valence Force Constants for Water from Vibrational Data, Paper Presented at the Thirtieth Symposium on Molecular Structure and Spectroscopy, Ohio State University, Columbus, OH.

is arbitrary and can be made to reduce the Hamiltonian $\underline{H}$[††] to contain only (n + 1) independent terms of total degree n in the components of total angular momentum for each even value of n. The standard choice is to remove the coefficients of operators such that the matrix elements in the symmetric-top basis satisfy the selection rule $\Delta K = 0, \pm 2$, that is, the reduced Hamiltonian matrix in a given vibrational state is at worst tridiagonal (Yallabandi and Parker,[7] Benedict et al[8]). This leads to the Watson reduced Hamiltonian

$$\begin{aligned}\underline{H}_w = {} & h_{000} + h_{100}\underline{P}^2 + h_{010}\underline{P}_z^2 + h_{001}(\underline{P}_x^2 - \underline{P}_y^2) \\ & + h_{200}\underline{P}^4 + h_{110}\underline{P}^2\underline{P}_z^2 + h_{020}\underline{P}_z^4 + h_{101}\{\underline{P}^2,(\underline{P}_x^2 - \underline{P}_y^2)\} \\ & + h_{011}\{\underline{P}_z^2,(\underline{P}_x^2 - \underline{P}_y^2)\} + h_{300}\underline{P}^6 + h_{210}\underline{P}^4\underline{P}_z^2 \\ & + h_{120}\underline{P}^2\underline{P}_z^4 + h_{030}\underline{P}_z^6 + h_{201}\{\underline{P}^4,(\underline{P}_x^2 - \underline{P}_y^2)\} \\ & + h_{111}\{\underline{P}^2\underline{P}_z^2,(\underline{P}_x^2 - \underline{P}_y^2)\} + h_{021}\{\underline{P}_z^4,(\underline{P}_x^2 - \underline{P}_y^2)\} \quad . \end{aligned} \tag{9}$$

Eq. (9) is given through fourth order, although many terms of high order in angular momentum can be preserved in the transformation Eq. (7) so that the equality is preserved. Notice that the choice of the parameters s_{pqr} has been made to eliminate any powers of the operator $(\underline{P}_x^2 - \underline{P}_y^2)$ greater than unity.

2.5 Direction Cosine Operators

Another example of the use of the commutator reduction is seen in the calculation of the dipole moment expansion (Clough et al,[9] Flaud et al[10]). This problem requires the calculation of commutators of the type

$$[(\underline{P}_x^p\underline{P}_y^q\underline{P}_z^r + \underline{P}_z^r\underline{P}_y^q\underline{P}_x^p),\ \underline{\Phi}_\alpha] \quad , \tag{10}$$

where the indices p, q, r are powers of angular momentum, and $\underline{\Phi}_\alpha$ is the direction cosine between the α-molecular-fixed axis and the space-fixed axis. The computer code COMMUTE generates series of reduced operators for the general

---

7. Yallabandi, K.Y. and Parker, P.M. (1968) J. Chem. Phys. 49:410.
8. Benedict, W.S., Clough, S.A., Frenkel, L., and Sullivan, T.E. (1970) J. Chem. Phys. 53:2565.
9. Clough, S.A., Beers, Y., Klein, G.P., and Rothman, L.S. (1973) J. Chem. Phys. 59:2254.
10. Flaud, J.-M., Camy-Peyret, C., Mandin, J.-Y., and Guelachvili, G. (1977) Mol. Phys. 34:413.

term (10) in a fashion similar to the results from the previous examples. The most general form of commutation operations involving the direction cosine operators occurs when performing a Watson-type transformation on the dipole moment expansion. The form of the commutator is then

$$[(\underline{P}_x^p \underline{P}_y^q \underline{P}_z^r + \underline{P}_z^r \underline{P}_y^q \underline{P}_x^p) \quad , \quad (\underline{P}_x^s \underline{P}_y^t \underline{P}_z^u \underline{\Phi}_\alpha + \underline{\Phi}_\alpha \underline{P}_z^u \underline{P}_y^t \underline{P}_x^s)] \tag{11}$$

Examples of the application of COMMUTE to terms (10) and (11) are given in Appendix B.4 and B.5.

References

1. Rothman, L.S. and Clough, S.A. (1971) J. Chem. Phys. 55:504.
2. Amat, G., Goldsmith, M., and Nielsen, H.H. (1957) J. Chem. Phys. 27:838.
3. Kneizys, F.X., Freedman, J.N., and Clough, S.A. (1966) J. Chem. Phys. 44:2552.
4. Watson, J.K.G. (1967) J. Chem. Phys. 46:1935; (1968) J. Chem. Phys. 48:4517.
5. Flaud, J.-M. and Camy-Peyret, C. (1973) Mol. Phys. 26:811.
6. Rothman, L.S. and Clough, S.A. (1975) Determination of Valence Force Constants for Water from Vibrational Data, Paper Presented at the Thirtieth Symposium on Molecular Structure and Spectroscopy, Ohio State University, Columbus, OH.
7. Yallabandi, K.Y. and Parker, P.M. (1968) J. Chem. Phys. 49:410.
8. Benedict, W.S., Clough, S.A., Frenkel, L., and Sullivan, T.E. (1970) J. Chem. Phys. 53:2565.
9. Clough, S.A., Beers, Y., Klein, G.P., and Rothman, L.S. (1973) J. Chem. Phys. 59:2254.
10. Flaud, J.-M., Camy-Peyret, C., Mandin, J.-Y., and Guelachvili, G. (1977) Mol. Phys. 34:413.

Appendix A

The Code COMMUTE

Provided in this Appendix is a complete listing of the program COMMUTE. Fundamental commutation relations are built in for the angular momentum operators and direction cosine operators. To distinguish the two types of operators, a general 6-vector was chosen with the correspondence: U → Φ_x, V → Φ_y, W → Φ_z, X → $\mathbf{P}_x$, Y → $\mathbf{P}_y$, and Z → $\mathbf{P}_z$.

The first input card indicates the number of terms in the initial polynomial that is to be reduced. However, the program automatically takes each initial term and forms an anticyclic pair to it. Thus, for the commutator $[\mathbf{P}_x, \mathbf{P}_y] = \mathbf{P}_x\mathbf{P}_y - \mathbf{P}_y\mathbf{P}_x$, it is only necessary to consider one term $\mathbf{P}_x\mathbf{P}_y$. The operator polynomial $[\{\mathbf{P}_x, \mathbf{P}_z\}, \mathbf{P}_y]$ should be considered as two terms for the program, $\mathbf{P}_x\mathbf{P}_z\mathbf{P}_y$ and $\mathbf{P}_z\mathbf{P}_x\mathbf{P}_y$; anticommutators should be expanded. This feature of doubling terms was undertaken since the primary use of the program requires commutator or anticommutator reduction. Single terms can nevertheless be handled as can be seen from the example in Appendix B. There is also the option on the first input card to allow the program to drive the reduction in a cyclic, that is, UVWXYZ, or anticyclic direction (ZYXWVU), thereby facilitating tests for axis invariance.

The second group of input cards (equal to the number of initial terms) indicates the number of operators, the coefficients, and the directions and exponents of the operators. The sample data deck accompanying the examples in Appendix B should suffice to explain the method of reading in the terms.

The program works on two "levels" at any time, a level being all terms of a given degree in exponent sum. The operators are scanned for their directions. If the directions are different, then the commutation relation is applied, if they are the same, then they are multiplied (exponents added). The action of the commutation relation creates a term at the next level. Terms at the second level are scanned for identical terms and cancelled when the coefficients are zero. When a level has been completely scanned and condensed, it is written as anti-commutators if the original term or terms was either hermitian or antihermitian. The process thus is one in which commutators are always "driven".

```
      PROGRAM COMMUTE(INPUT,OUTPUT)                                          COM   10
C                                                                            COM   11
C                   PROGRAM BY LAURENCE S. ROTHMAN                           COM   12
C                   OPTICAL PHYSICS DIVISION                                 COM   13
C                   AIR FORCE GEOPHYSICS LABORATORY                          COM   14
C                   BEDFORD, MA  01731                                       COM   15
C                                                                            COM   16
C THIS PROGRAM DRIVES TERMS CONTAINING ANGULAR MOMENTUM AND/OR               COM   20
C DIRECTION COSINE OPERATORS INTO EITHER CYCLIC OR ANTICYCLIC TERMS          COM   21
C (ANTICOMMUTATORS) THROUGH THE USE OF COMMUTATION RELATIONS.                COM   22
C                                                                            COM   25
C FOR THE ARRAYS IN BLANK COMMON:                                            COM   30
C FIRST INDEX DESIGNATES LEVEL, SECOND:TERM, THIRD:POSITION OF OPERATOR.     COM   40
C IFOSMAX(LEV,ITERM) = NUMBER OF OPERATORS IN ITERM.                         COM   50
C ITERML(LEV) = TOTAL NUMBER OF TERMS IN LEV.                                COM   60
C PROGRAM WILL TERMINATE IF IT ENCOUNTERS: (1) TWELVE OR MORE OPERATORS      COM   70
C IN A TERM, OR (2) 200 OR MORE TERMS AT A PARTICULAR LEVEL.                 COM   80
C                                                                            COM   90
      COMMON IOPS(2,199,11),IEXP(2,199,11),COEFF(2,199),IFOSMAX(2,199)       COM  100
      DIMENSION ITERML(2), DIR(6), X(11), X2(11), EPS(6,6), ICTEL(6,6),      COM  110
     1HEXP(11), IPIC(9), HEXP2(11), EPSVEC(36), ICVEC(36)                    COM  120
      DATA (DIR(I),I=1,6)/1HU,1HV,1HW,1HX,1HY,1HZ/                           COM  130
      DATA (HDIG(I),I=1,9)/1H1,1H2,1H3,1H4,1H5,1H6,1H7,1H8,1H9/              COM  140
      DATA HP,HLP,HRP,HPL,HMI,H9/1HP,1H(,1H),1H+,1H-,1H /                    COM  150
C                                                                            COM  160
C COMMUTATION RELATIONS:                                                     COM  170
C                                                                            COM  180
      DATA EPSVEC/
C           U          V          W          X          Y          Z
     U      0. ,       0. ,       0. ,       0. ,      -1. ,       1  ,
     V      0. ,       0. ,       0. ,       1. ,       0. ,      -1. ,
     W      0. ,       0. ,       0. ,      -1. ,       1. ,       0. ,
     X      0. ,      -1. ,       1. ,       0. ,      -1. ,       1. ,
     Y      1. ,       0. ,      -1. ,       1. ,       0. ,      -1. ,
     Z     -1. ,       1. ,       0. ,      -1. ,       1. ,       0. /
C
      DATA ICVEC/
C           U          V          W          X          Y          Z
     U      0  ,       5  ,       0  ,       0  ,       3  ,       2  ,
     V      0  ,       0  ,       0  ,       3  ,       0  ,       1  ,
     W      0  ,       0  ,       0  ,       2  ,       1  ,       0  ,
     X      5  ,       3  ,       2  ,       0  ,       6  ,       5  ,
     Y      3  ,       0  ,       1  ,       6  ,       0  ,       4  ,
     Z      2  ,       1  ,       0  ,       5  ,       4  ,       0  /
C
      DO 10 I=1,6
      DO 10 J=1,6
      IND=J+6*(I-1)
      EPS(I,J)=EPSVEC(IND)
      ICTEL(I,J)=ICVEC(IND)
10    CONTINUE                                                               COM  310
C                                                                            COM  320
C (1) READ IN NUMBER OF INPUT TERMS (I2) AND WHETHER TERMS ARE TO BE         COM  330
C ORTHEN CYCLICALLY OR ANTICYCLICALLY (F6.1):                                COM  340
C (2) READ NUMBER OF OPERATORS (I2), COEFFICIENTS (2F6.1), AND DIRECTIONS    COM  350
C AND EXPONENT OF EACH OPERATOR -- FORMAT (I2,2F6.1,11(1X,A1,I2,2X)):        COM  360
C                                                                            COM  370
```

```
      LEV=1                                                                 COM  380
      LEV2=2                                                                COM  390
20    READ 900, ITERML(1),CYCLE                                             COM  400
      CYC=10HCYCLIC                                                         COM  410
      IF (CYCLE .LT. -0.1) CYC=10HANTICYCLIC                                COM  420
      ITERML(1)=2*ITERML(1)                                                 COM  430
      TTL=ITERML(1)                                                         COM  440
      CALL SECOND (TIME)                                                    COM  450
      PRINT 905, TIME,CYC                                                   COM  460
      IF (TTL.EQ.0) STOP                                                    COM  470
      DO 60 IT=1,TTL,2                                                      COM  480
      READ 900, MAX,COEFF(1,IT),COEFF(1,IT+1),(X(IP),IEXP(1,IT,IP),IP=1,CCM  490
     1MAX)                                                                  COM  500
      IPOSMAX(1,IT)=IPOSMAX(1,IT+1)=MAX                                     COM  510
      DO 30 I=1,MAX                                                         COM  520
      DO 30 J=1,8                                                           COM  530
      IF (DIP(J).EQ.X(I)) IDIR(1,IT,I)=J                                    COM  540
30    CONTINUE                                                              COM  550
C                                                                           COM  560
C AUTOMATICALLY SYMMETRIZE INPUT TERMS:                                     COM  570
C                                                                           COM  580
      ISUM=0                                                                COM  590
      DO 40 I=1,MAX                                                         COM  600
      ISUM=ISUM+IEXP(1,IT,I)                                                COM  610
      J=MAX+1-I                                                             COM  620
      IDIR(1,IT+1,I)=IDIR(1,IT,J)                                           COM  630
      IEXP(1,IT+1,I)=IEXP(1,IT,J)                                           COM  640
      NI=IEXP(1,IT,I)                                                       COM  650
      NJ=IEXP(1,IT+1,I)                                                     COM  660
      HEXP(I)=HDIG(NI)                                                      COM  670
      HEXP2(I)=HDIG(NJ)                                                     COM  680
40    X2(I)=X(J)                                                            COM  690
C                                                                           COM  700
C TEST FOR HERMITICITY, -1 FOR NON-HERMITIAN INPUT OPERATOR,                COM  710
C 0 FOR HERMITIAN OPERATOR, +1 FOR ANTI-HERMITIAN OPERATOR:                 COM  720
C                                                                           COM  730
      IRAT=COEFF(1,IT+1)/COEFF(1,IT)+1.1                                    COM  740
      IRAR=ISUM-2*(ISUM/2)                                                  COM  750
      IHERM=1                                                               COM  760
      IF ((IRAR.EQ.1).AND.(IRAT.EQ.0)) IHERM=0                              COM  770
      IF ((IRAR.EQ.0).AND.(IRAT.EQ.2)) IHERM=0                              COM  780
      IF (IRAT.EQ.1) IHERM=-1                                               COM  790
C                                                                           COM  800
C PRINT INPUT TERMS:                                                        COM  810
C                                                                           COM  820
      IF ((MAX.EQ.1).OR.(IHERM.EQ.-1)) GO TO 50                             COM  830
      PRINT 940, (HEXP(IP),HB,IP=1,MAX),HB,HB,HB,(HEXP2(IF),HB,IF=1,MAX)COM  840
      XSIGN=H+T                                                             COM  850
      IF (IRAT.EQ.2) XSIGN=HPL                                              COM  860
      CALL FRACT (COEFF(1,IT),IC,ND)                                        COM  870
      IF (ND.EQ.0) PRINT 910, IC,HB,HLP,HB,(HP,HB,IP=1,MAX),HB,XSIGN,       COM  880
     1(HB,HP,IP=1,MAX),HB,HRP                                               COM  890
      IF ((ND.GT.0).AND.(ND.LE.64)) PRINT 920, IC,ND,HB,HLF,HB,(HP,HB,      COM  900
     1IP=1,MAX),HB,XSIGN,(HB,HP,IP=1,MAX),HB,HRF                            COM  910
      IF (ND.GT.64) PRINT 930, COEFF(1,IT),HB,HLP,HB,(HP,HB,IP=1,MAX),HBCOM  920
     1,XSIGN,(HB,HP,IP=1,MAX),HB,HRP                                        COM  930
      PRINT 940, (X(IP),HB,IP=1,MAX),HB,HB,HB,(X2(IF),HB,IF=1,MAX)         COM  940
```

```
      GO TO 60                                                          COM  950
50    PRINT 940, (HFXP(IP),HB,IP=1,MAX)                                 COM  960
      C=COEFF(1,IT)+COEFF(1,IT+1)                                       COM  970
      CALL FRACT (C,IC,ND)                                              COM  980
      IF (ND.EQ.0) PRINT 910, IC,HB,HB,HB,(HP,HB,IP=1,MAX)              COM  990
      IF ((ND.GT.0).AND.(ND.LE.64)) PRINT 920, IC,ND,HB,HB,HB,(HP,HB,   COM 1000
     1IP=1,MAX)                                                         COM 1010
      IF (ND.GT.64) PRINT 930, C,HB,HB,HB,(HP,HB,IP=1,MAX)              COM 1020
      PRINT 940, (X(IP),HB,IP=1,MAX)                                    COM 1030
60    CONTINUE                                                          COM 1040
      PRINT 950                                                         COM 1050
C                                                                       COM 1060
C INDEX LEVELS, TERMS, AND TEST POSITION OF OPERATORS (SHIFT LEVELS TO  COM 1070
C SAVE STORAGE):                                                        COM 1080
C                                                                       COM 1090
      L=0                                                               COM 1100
70    L=L+1                                                             COM 1110
      IF (L.EQ.1) GO TO 90                                              COM 1120
      ITL=ITERML(LEV2)                                                  COM 1130
      IF (ITL.EQ.0) GO TO 20                                            COM 1140
      ITERML(LEV)=ITL                                                   COM 1150
      DO 80 IT=1,ITL                                                    COM 1160
      COEFF(LEV,IT)=COEFF(LEV2,IT)                                      COM 1170
      MAX=IPOSMAX(LEV2,IT)                                              COM 1180
      IPOSMAX(LEV,IT)=MAX                                               COM 1190
      DO 60 IP=1,MAX                                                    COM 1200
      IDIR(LEV,IT,IP)=IDIR(LEV2,IT,IP)                                  COM 1210
80    IEXP(LEV,IT,IP)=IEXP(LEV2,IT,IP)                                  COM 1220
90    ITERML(LEV2)=ITERM=ITERM2=0                                       COM 1230
100   ITERM=ITERM+1                                                     COM 1240
      IF (ITERML(LEV).EQ.0) GO TO 70                                    COM 1250
      IF (ITERM.GT.ITERML(LEV)) GO TO 340                               COM 1260
110   IPOS=0                                                            COM 1270
120   IPOS=IPOS+1                                                       COM 1280
      IF (IPOS.GE.IPOSMAX(LEV,ITERM)) GO TO 390                         COM 1290
      N=IPOS+1                                                          COM 1300
      IF (IDIR(LEV,ITERM,N).EQ.IDIR(LEV,ITERM,IPOS)) GO TO 130          COM 1310
      IF (CYCLE .LT. -0.1) GO TO 121                                    COM 1320
      IF (IDIR(LEV,ITERM,IPOS).GT.IDIR(LEV,ITERM,N)) GO TO 160          COM 1330
      GO TO 120                                                         COM 1340
121   IF (IDIR(LEV,ITERM,IPOS).LT.IDIR(LEV,ITERM,N)) GO TO 160          COM 1350
      GO TO 120                                                         COM 1360
C                                                                       COM 1370
C CONTRACT ADJACENT OPERATORS HAVING IDENTICAL DIRECTIONS, IN LEV:      COM 1380
C                                                                       COM 1390
130   IEXP(LEV,ITERM,IPOS)=IEXP(LEV,ITERM,IPOS)+IEXP(LEV,ITERM,N)       COM 1400
      MAX=IPOSMAX(LEV,ITERM)-1                                          COM 1410
      IF (MAX.LT.N) GO TO 150                                           COM 1420
      DO 140 I=N,MAX                                                    COM 1430
      IDIR(LEV,ITERM,I)=IDIR(LEV,ITERM,I+1)                             COM 1440
140   IEXP(LEV,ITERM,I)=IEXP(LEV,ITERM,I+1)                             COM 1450
150   IPOSMAX(LEV,ITERM)=MAX                                            COM 1460
      GO TO 110                                                         COM 1470
C                                                                       COM 1480
C COMMUTE TWO OPERATORS IN LEV:                                         COM 1490
C                                                                       COM 1500
160   IA=IDIR(LEV,ITERM,IPOS)                                           COM 1510
```

```
      I3=IDIR(LEV,ITERM,N)                                              COM 1520
      IEXP(LEV,ITERM,IPOS)=IEXP(LEV,ITERM,IPOS)-1                       COM 1530
      IPOSMAX(LEV,ITERM)=IPOSMAX(LEV,ITERM)+2                           COM 1540
      MAX=IPOSMAX(LEV,ITERM)                                            COM 1550
      IF (MAX.LE.11) GO TO 170                                          COM 1560
      PRINT 970, MAX,L,ITERM                                            COM 1570
      GO TO 20                                                          COM 1580
170   LIM=MAX-N                                                         COM 1590
      DO 180 I=1,LIM                                                    COM 1600
      J=MAX+1-I                                                         COM 1610
      IDIR(LEV,ITERM,J)=IDIR(LEV,ITERM,J-2)                             COM 1620
180   IEXP(LEV,ITERM,J)=IEXP(LEV,ITERM,J-2)                             COM 1630
      IEXP(LEV,ITERM,N+2)=IEXP(LEV,ITERM,N)-1                           COM 1640
      IEXP(LEV,ITERM,N+1)=IEXP(LEV,ITERM,N)+1                           COM 1650
C                                                                       COM 1660
C ELIMINATE ZERO-EXPONENTS IN LEV:                                      COM 1670
C                                                                       COM 1680
      N2=N+2                                                            COM 1690
      NGAM=N                                                            COM 1700
      IF (IEXP(LEV,ITERM,N2).GT.0) GO TO 210                            COM 1710
      MAX=IPOSMAX(LEV,ITERM)-1                                          COM 1720
      IF (MAX.LT.N2) GO TO 200                                          COM 1730
      DO 190 I=N2,MAX                                                   COM 1740
      IDIR(LEV,ITERM,I)=IDIR(LEV,ITERM,I+1)                             COM 1750
190   IEXP(LEV,ITERM,I)=IEXP(LEV,ITERM,I+1)                             COM 1760
200   IPOSMAX(LEV,ITERM)=MAX                                            COM 1770
210   IF (IEXP(LEV,ITERM,IPOS).GT.0) GO TO 230                          COM 1780
      NGAM=NGAM-1                                                       COM 1790
      MAX=IPOSMAX(LEV,ITERM)-1                                          COM 1800
      DO 220 I=IPOS,MAX                                                 COM 1810
      IDIR(LEV,ITERM,I)=IDIR(LEV,ITERM,I+1)                             COM 1820
220   IEXP(LEV,ITERM,I)=IEXP(LEV,ITERM,I+1)                             COM 1830
      IPOSMAX(LEV,ITERM)=MAX                                            COM 1840
230   IF (EPS(IA,IB).EQ.0.) GO TO 110                                   COM 1850
C                                                                       COM 1860
C CREATE TERM2 IN LEV2 (= LEV+1):                                       COM 1870
C                                                                       COM 1880
      ITERM2=ITERM2+1                                                   COM 1890
      IF (ITERM2.LE.199) GO TO 240                                      COM 1900
      PRINT 980, ITERM2,L,ITERM                                         COM 1910
      GO TO 20                                                          COM 1920
240   ITERML(LEV2)=ITERM2                                               COM 1930
      IPOSMAX(LEV2,ITERM2)=IPOSMAX(LEV,ITERM)-1                         COM 1940
      IP=NGAM-1                                                         COM 1950
      IF (IP.LT.1) GO TO 260                                            COM 1960
      DO 250 I=1,IP                                                     COM 1970
      IDIR(LEV2,ITERM2,I)=IDIR(LEV,ITERM,I)                             COM 1980
250   IEXP(LEV2,ITERM2,I)=IEXP(LEV,ITERM,I)                             COM 1990
260   COEFF(LEV2,ITERM2)=EPS(IA,IB)*COEFF(LEV,ITERM)                    COM 2000
      IDIR(LEV2,ITERM2,NGAM)=ICTBL(IA,IB)                               COM 2010
      IEXP(LEV2,ITERM2,NGAM)=1                                          COM 2020
      MAX=IPOSMAX(LEV2,ITERM2)                                          COM 2030
      N1=NGAM+1                                                         COM 2040
      IF (MAX.LT.N1) GO TO 280                                          COM 2050
      DO 270 I=N1,MAX                                                   COM 2060
      IDIR(LEV2,ITERM2,I)=IDIR(LEV,ITERM,I+1)                           COM 2070
270   IEXP(LEV2,ITERM2,I)=IEXP(LEV,ITERM,I+1)                           COM 2080
```

```
C                                                                       COM 2090
C CONTRACT ADJACENT OPERATORS HAVING IDENTICAL DIRECTIONS IN LEV2:       COM 2100
C                                                                       COM 2110
280   IF (MAX.LT.2) GO TO 310                                           COM 2120
      LMAX=MAX                                                          COM 2130
      LI=1                                                              COM 2140
      DO 300 I=2,MAX                                                    COM 2150
      LI=LI+1                                                           COM 2160
      IF (LMAX.LT.LI) GO TO 310                                         COM 2170
      IF (IDIR(LEV2,ITERM2,LI).NE.IDIR(LEV2,ITERM2,LI-1)) GO TO 300     COM 2180
      IEXP(LEV2,ITERM2,LI-1)=IEXP(LEV2,ITERM2,LI-1)+IEXP(LEV2,ITERM2,LI)COM 2190
      LMAX=LMAX-1                                                       COM 2200
      IPOSMAX(LEV2,ITERM2)=LMAX                                         COM 2210
      IF (LMAX.LT.LI) GO TO 310                                         COM 2220
      DO 290 J=LI,LMAX                                                  COM 2230
      IDIR(LEV2,ITERM2,J)=IDIR(LEV2,ITERM2,J+1)                         COM 2240
290   IEXP(LEV2,ITERM2,J)=IEXP(LEV2,ITERM2,J+1)                         COM 2250
      LI=LI-1                                                           COM 2260
300   CONTINUE                                                          COM 2270
C                                                                       COM 2280
C ADD IDENTICAL TERMS IN LEV2:                                          COM 2290
C                                                                       COM 2300
310   IF (ITERM2.LE.1) GO TO 110                                        COM 2310
      MAX=IPOSMAX(LEV2,ITERM2)                                          COM 2320
      K=ITERM2-1                                                        COM 2330
      DO 330 I=1,K                                                      COM 2340
      IF (IPOSMAX(LEV2,I).NE.MAX) GO TO 330                             COM 2350
      DO 320 J=1,MAX                                                    COM 2360
      IF (IDIR(LEV2,I,J).NE.IDIR(LEV2,ITERM2,J)) GO TO 330              COM 2370
      IF (IEXP(LEV2,I,J).NE.IEXP(LEV2,ITERM2,J)) GO TO 330              COM 2380
320   CONTINUE                                                          COM 2390
      COEFF(LEV2,I)=COEFF(LEV2,I)+COEFF(LEV2,ITERM2)                    COM 2400
      ITERM2=ITERM2-1                                                   COM 2410
      ITERML(LEV2)=ITERM2                                               COM 2420
      CALL CANCEL (LEV2,ITERM2,I,ITERML(LEV2))                          COM 2430
      GO TO 110                                                         COM 2440
330   CONTINUE                                                          COM 2450
      GO TO 110                                                         COM 2460
C                                                                       COM 2470
C TEST FOR ANTI-COMMUTATORS OR COMMUTATORS:                             COM 2480
C                                                                       COM 2490
340   IF (IHERM.EQ.-1) GO TO 460                                        COM 2500
      ISUM=0                                                            COM 2510
      MAX=IPOSMAX(LEV,1)                                                COM 2520
      DO 350 I=1,MAX                                                    COM 2530
350   ISUM=ISUM+IEXP(LEV,1,I)                                           COM 2540
      IPAR=ISUM-2*(ISUM/2)                                              COM 2550
      IF (IHERM.EQ.IPAR) GO TO 460                                      COM 2560
      ITL=ITERML(LEV)                                                   COM 2570
      IF (ITL.LT.100) GO TO 360                                         COM 2580
      PRINT 990, ITL,L,ITERM                                            COM 2590
      GO TO 20                                                          COM 2600
360   DO 380 IT=1,ITL                                                   COM 2610
      MAX=IPOSMAX(LEV,IT)                                               COM 2620
      IPOSMAX(LEV,IT+ITL)=MAX                                           COM 2630
      COEFF(LEV,IT)=0.5*COEFF(LEV,IT)                                   COM 2640
      COEFF(LEV,IT+ITL)=-COEFF(LEV,IT)                                  COM 2650
```

```
      DO 370 I=1,MAX                                                    COM 2660
      J=MAX+1-I                                                         COM 2670
      IEXP(LEV,IT+ITL,J)=IEXP(LEV,IT,I)                                 COM 2680
370   IDIR(LEV,IT+ITL,J)=IDIR(LEV,IT,I)                                 COM 2690
380   CONTINUE                                                          COM 2700
      ITERM=ITL+1                                                       COM 2710
      ITERML(LEV)=2*ITL                                                 COM 2720
      GO TO 110                                                         COM 2730
C                                                                       COM 2740
C ADD IDENTICAL TERMS IN LEV:                                           COM 2750
C                                                                       COM 2760
390   IF (ITERM.EQ.1) GO TO 440                                         COM 2770
      MAX=IPOSMAX(LEV,ITERM)                                            COM 2780
      K=ITERM-1                                                         COM 2790
      DO 430 I=1,K                                                      COM 2800
      IF (IPOSMAX(LEV,I).NE.MAX) GO TO 430                              COM 2810
      DO 400 J=1,MAX                                                    COM 2820
      IF (IDIR(LEV,I,J).NE.IDIR(LEV,ITERM,J)) GO TO 430                 COM 2830
      IF (IEXP(LEV,I,J).NE.IEXP(LEV,ITERM,J)) GO TO 430                 COM 2840
400   CONTINUE                                                          COM 2850
      COEFF(LEV,I)=COEFF(LEV,I)+COEFF(LEV,ITERM)                        COM 2860
      ITERML(LEV)=ITERML(LEV)-1                                         COM 2870
      ITL=ITERML(LEV)                                                   COM 2880
      IF (ITL.LT.ITERM) GO TO 420                                       COM 2890
      DO 410 IT=ITERM,ITL                                               COM 2900
      LIM=IPOSMAX(LEV,IT+1)                                             COM 2910
      COEFF(LEV,IT)=COEFF(LEV,IT+1)                                     COM 2920
      IPOSMAX(LEV,IT)=LIM                                               COM 2930
      DO 410 IP=1,LIM                                                   COM 2940
      IDIR(LEV,IT,IP)=IDIR(LEV,IT+1,IP)                                 COM 2950
410   IEXP(LEV,IT,IP)=IEXP(LEV,IT+1,IP)                                 COM 2960
420   ITERM=ITERM-1                                                     COM 2970
      GO TO 450                                                         COM 2980
430   CONTINUE                                                          COM 2990
      GO TO 100                                                         COM 3000
C                                                                       COM 3010
C CANCEL ZERO-COEFFICIENT TERMS IN LEV:                                 COM 3020
C                                                                       COM 3030
440   I=1                                                               COM 3040
450   CALL CANCEL (LEV,ITERM,I,ITERML(LEV))                             COM 3050
      GO TO 100                                                         COM 3060
C                                                                       COM 3070
C WRITE ANSWERS:                                                        COM 3080
C                                                                       COM 3090
460   ITL=ITERML(LEV)                                                   COM 3100
      IF (ITL.EQ.0) GO TO 70                                            COM 3110
      IF (L.EQ.1) GO TO 470                                             COM 3120
      LL=L-1                                                            COM 3130
      PRINT 950, LL                                                     COM 3140
470   DO 500 IT=1,ITL                                                   COM 3150
      IF (ABS(COEFF(LEV,IT)).LT.1.E-20) GO TO 500                       COM 3160
      MAX=IPOSMAX(LEV,IT)                                               COM 3170
      DO 480 I=1,MAX                                                    COM 3180
      J=MAX+1-I                                                         COM 3190
      NI=IDIR(LEV,IT,I)                                                 COM 3200
      X(I)=DIR(NI)                                                      COM 3210
      NII=IEXP(LEV,IT,I)                                                COM 3220
```

```
      HEXP(I)=HOLD(NII)                                                    COM 3230
      X2(J)=X(I)                                                           COM 3240
480   HEXP2(J)=HEXP(I)                                                     COM 3250
      IF ((MAX.EQ.1).OR.(ITERM.EQ.-1)) GO TO 490                           COM 3260
      PRINT 940, (HEXP(IP),HB,IP=1,MAX),HB,HB,HE,(HEXP2(IP),HE,IP=1,MAX)COM 3270
      C=COEFF(LEV,TT)/2.                                                   COM 3280
      CALL FRACT (C,IC,ND)                                                 COM 3290
      IF (ND.EQ.0) PRINT 910, IC,HB,HLP,HB,(HP,HB,IP=1,MAX),HB,HPL,       COM 3300
     1(HB,HP,IP=1,MAX),HB,HRP                                              COM 3310
      IF ((ND.GT.0).AND.(ND.LE.64)) PRINT 920, IC,ND,HB,HLP,HB,(HP,HB,     COM 3320
     1IP=1,MAX),HB,HPL,(HB,HP,IP=1,MAX),HB,HRP                             COM 3330
      IF (ND.GT.64) PRINT 930, C,HB,HLP,HB,(HP,HB,IP=1,MAX),HB,HPL,       COM 3340
     1(HB,HP,IP=1,MAX),HB,HRP                                              COM 3350
      PRINT 940, (X(IP),HB,IP=1,MAX),HB,HB,HB,(X2(IP),HB,IP=1,MAX)        COM 3360
      GO TO 500                                                            COM 3370
490   PRINT 940, (HEXP(IP),HB,IP=1,MAX)                                   COM 3380
      CALL FRACT (COEFF(LEV,IT),IC,ND)                                     COM 3390
      IF (ND.EQ.0) PRINT 910, IC,HB,HB,HB,(HP,HB,IP=1,MAX)                COM 3400
      IF ((ND.GT.0).AND.(ND.LE.64)) PRINT 920, IC,ND,HB,HB,HB,(HP,HB,     COM 3410
     1IP=1,MAX)                                                            COM 3420
      IF (ND.GT.64) PRINT 930, COEFF(LEV,IT),HB,HE,HB,(HP,HE,IP=1,MAX)    COM 3430
      PRINT 940, (X(IP),HB,IP=1,MAX)                                      COM 3440
500   CONTINUE                                                             COM 3450
      GO TO 70                                                             COM 3460
C                                                                          COM 3470
900   FORMAT (I2,2F6.1,11(1X,A1,I2,2X))                                    COM 3480
905   FORMAT (30H1 ********** THE TIME IS NOW =,F9.3,19H SECONDS *******COM 3490
     1***5X,A10/)                                                          COM 3500
910   FORMAT (14X,I6,80A1)                                                 COM 3510
920   FORMAT (11X,I6,"/",I2,80A1)                                          COM 3520
930   FORMAT (10X,F12.6,80A1)                                              COM 3530
940   FORMAT (26X,80A1)                                                    COM 3540
950   FORMAT (1H0,24X"EQUALS"/)                                            COM 3550
960   FORMAT (1H0/1H0,24X,I2/23X,54+I* */)                                 COM 3560
970   FORMAT (" ABORTED:IPOSMAX(LEV,ITERM)=",3I6)                          COM 3570
980   FORMAT (" ABORTED:ITERM2=",3I6)                                      COM 3580
990   FORMAT (" ABORTED:ITERML(LEV)=",3I6)                                 COM 3590
      END                                                                  COM 3600
```

```
      SUBROUTINE CANCEL (L,ITERM,ITE,ITL)                                CAN  10
C THIS SUBROUTINE                                                        CAN  14
C ANNIHILATES A TERM WHOSE COEFFICIENT IS LESS THAN 1.E-12.              CAN  15
      COMMON IDIF(2,199,11),IEXP(2,199,11),COEFF(2,199),IPOSMAX(2,199)   CAN  20
      IF (ABS(COEFF(L,ITE)).GT.1.E-12) RETURN                            CAN  30
      ITERM=ITERM-1                                                      CAN  40
      ITL=ITL-1                                                          CAN  50
      IF (ITL.LT.ITE) RETURN                                             CAN  60
      DO 10 IT=ITE,ITL                                                   CAN  70
      MAX=IPOSMAX(L,IT+1)                                                CAN  80
      COEFF(L,IT)=COEFF(L,IT+1)                                          CAN  90
      IPOSMAX(L,IT)=MAX                                                  CAN 100
      DO 10 IP=1,MAX                                                     CAN 110
      IDIF(L,IT,IP)=IDIF(L,IT+1,IP)                                      CAN 120
10    IEXP(L,IT,IP)=IEXP(L,IT+1,IP)                                      CAN 130
      RETURN                                                             CAN 140
      END                                                                CAN 150-
```

```
      SUBROUTINE FRACT (C,IC,ND)                                         FRA  10
C THIS SUBROUTINE ATTEMPTS TO WRITE COEFFICIENTS AS INTEGERS, OR         FRA  15
C FRACTIONS IF THEY ARE AT LEAST MULTIPLES OF 1/64.                      FRA  16
      ND=0                                                               FRA  20
      XC=C+SIGN(1.E-08,C)                                                FRA  30
      IC=XC                                                              FRA  40
      DIFC=ABS(XC-FLOAT(IC))                                             FRA  50
      IF (DIFC.LT.1.E-06) RETURN                                         FRA  60
      DO 10 I=1,7                                                        FRA  70
      ND=2**I                                                            FRA  80
      DIF=FLOAT(ND)*DIFC                                                 FRA  90
      IDIF=DIF+1.E-08                                                    FRA 100
      IF (DIF-FLOAT(IDIF).LT.1.E-06) GO TO 20                            FRA 110
10    CONTINUE                                                           FRA 120
      RETURN                                                             FRA 130
20    IC=ND*IC+ISIGN(IDIF,IC)                                            FRA 140
      RETURN                                                             FRA 150
      END                                                                FRA 160-
```

Appendix B

EXAMPLES

B.1 COMMUTATORS ARISING FROM DIAGONALIZATION OF HAMILTONIAN

$[\{P_x, P_z\}], P_y^2]$

Input cards:

```
2
3   0.5   -0.5 X 1   Z 1   Y 2
3   0.5   -0.5 Z 1   X 1   Y 2
```

Result:

```
          1 1 2     2 1 1
1/ 2 ( P P P   - P P P )
          X Z Y     Y Z X
          1 1 2     2 1 1
1/ 2 ( P P P   - P P P )
          Z X Y     Y X Z

        EQUALS


          1
      +IM *

          2 1     1 2
   1 ( P P   + P P )
          X Y     Y X
          1 2     2 1
  -1 ( P P   + P P )
          Y Z     Z Y
```

$[\{P_x^2, P_y^2\}, \{P_x, P_z\}]$

Input cards:

```
4
4  0.25 -0.25 X 2   Y 2   X 1   Z 1
4  0.25 -0.25 X 2   Y 2   Z 1   X 1
4  0.25 -0.25 Y 2   X 2   X 1   Z 1
4  0.25 -0.25 Y 2   X 2   Z 1   X 1
```

Result:

```
********** THE TIME IS NOW =     4.268 SECONDS **********      CYCLIC
```

$$
\begin{array}{l}
 {\scriptstyle 2\ 2\ 1\ 1} \qquad {\scriptstyle 1\ 1\ 2\ 2} \\
1/4\ (\ P_{XY}\,P_{XY}\,P_{XZ}\,P_{YX} - P_{XY}\,P_{XZ}\,P_{YX}\,P_{YX}\ ) \\
 {\scriptstyle 2\ 2\ 1\ 1} \qquad {\scriptstyle 1\ 1\ 2\ 2} \\
1/4\ (\ P_{XY}\,P_{XY}\,P_{ZX}\,P_{YX} - P_{XY}\,P_{ZX}\,P_{YX}\,P_{YX}\ ) \\
 {\scriptstyle 2\ 2\ 1\ 1} \qquad {\scriptstyle 1\ 1\ 2\ 2} \\
1/4\ (\ P_{YX}\,P_{XY}\,P_{XZ}\,P_{YX} - P_{XY}\,P_{XZ}\,P_{YX}\,P_{YX}\ ) \\
 {\scriptstyle 2\ 2\ 1\ 1} \qquad {\scriptstyle 1\ 1\ 2\ 2} \\
1/4\ (\ P_{YX}\,P_{YX}\,P_{ZX}\,P_{YX} - P_{XY}\,P_{ZX}\,P_{YX}\,P_{YX}\ )
\end{array}
$$

EQUALS

$$
\begin{array}{l}
 {\scriptstyle 1} \\
+IM\ * \\
 {\scriptstyle 2\ 3} \qquad {\scriptstyle 3\ 2} \\
1\ (\ P_{XY}\,P_{XY} + P_{XY}\,P_{YX}\ ) \\
 {\scriptstyle 4\ 1} \qquad {\scriptstyle 1\ 4} \\
-1\ (\ P_{XY}\,P_{XY} + P_{XY}\,P_{YX}\ ) \\
 {\scriptstyle 2\ 1\ 2} \qquad {\scriptstyle 2\ 1\ 2} \\
1\ (\ P_{XY}\,P_{XY}\,P_{Z} + P_{XY}\,P_{Z}\,P_{YX}\ )
\end{array}
$$

$$
\begin{array}{l}
 {\scriptstyle 3} \\
+IM\ * \\
 {\scriptstyle 2\ 1} \qquad {\scriptstyle 1\ 2} \\
1\ (\ P_{XY}\,P_{XY} + P_{XY}\,P_{YX}\ ) \\
 {\scriptstyle 3} \\
1/2\ \ P_{Y} \\
 {\scriptstyle 1\ 2} \qquad {\scriptstyle 2\ 1} \\
-1\ (\ P_{YZ}\,P_{YZ} + P_{ZY}\,P_{ZY}\ )
\end{array}
$$

B.2 REDUCTION OF SINGLE TERM

$P_x P_y P_x P_y P_z P_z$

Input cards:

```
1
6   1.0    0.0 X 1   Y 1   X 1   Y 1   Z 1   Z 1
```

Result:

```
********** THE TIME IS NOW =     4.403 SECONDS **********      CYCLIC:

                      1 1 1 1 1 1
                1     P P P P P P
                      X Y X Y Z Z

                    EQUALS

                      2 2 2
                1     P P P
                      X Y Z


                       1
                    +IM *

                      1 1 3
                1     P P P
                      X Y Z


                       2
                    +IM *

                      2 2
                1     P P
                      X Z
```

B.3 COMMUTATOR ARISING FROM TRANSFORMATION TO MINIMAL SET

$$[\tfrac{1}{2}(P_x P_y P_z + P_z P_y P_x),\ \tfrac{1}{2}\{P_x^2, P_y^2\}]$$

Input Cards:

```
4
5  0.25 -0.25 X 1   Y 1   Z 1   X 2   Y 2
5  0.25 -0.25 X 1   Y 1   Z 1   Y 2   X 2
5  0.25 -0.25 Z 1   Y 1   X 1   X 2   Y 2
5  0.25 -0.25 Z 1   Y 1   X 1   Y 2   X 2
```

Result:

```
          1 1 1 2 2     2 2 1 1 1
1/ 4 ( P P P P P   - P P P P P )
          X Y Z X Y     Y X Z Y X
          1 1 1 2 2     2 2 1 1 1
1/ 4 ( P P P P P   - P P P P P )
          X Y Z Y X     X Y Z Y X
          1 1 1 2 2     2 2 1 1 1
1/ 4 ( P P P P P   - P P P P P )
          Z Y X X Y     Y X X Y Z
          1 1 1 2 2     2 2 1 1 1
1/ 4 ( P P P P P   - P P P P P )
          Z Y X Y X     X Y X Y Z

        EQUALS

          1
      +IM *

          2 4     4 2
  -1 ( P P   + P P )
          X Y     Y X
          4 2     2 4
   1 ( P P   + P P )
          X Y     Y X


          3
      +IM *

          4
   2    P
          X
          2 2     2 2
  -5 ( P P   + P P )
          X Z     Z X
          4
  -2    P
          Y
          2 2     2 2
   5 ( P P   + P P )
          Y Z     Z Y
```

```
      5
   +IM *

      2
-8    P
       X
      2
 8    P
       Y
```

B.4 DIPOLE MOMENT COMMUTATOR

$$[P_y^2, \Phi_x]$$

Input cards:

```
1
2   1.0   -1.0 Y 2    U 1
```

Result:

```
       2 1      1 2
1 ( P P   - P P   )
       Y U      U Y

     EQUALS
```

```
      1
   +IM *

       1 1      1 1
1 ( P P   + P P   )
       W Y      Y W
```

B.5 TRANSFORMING INTENSITIES

$$[\tfrac{1}{2}(P_xP_yP_z + P_zP_yP_x)\,,\ \tfrac{1}{2}\{P_x^2, \Phi_z\}]$$

Input cards:

```
4
5   0.25 -0.25 X 1    Y 1    Z 1    X 2    W 1
5   0.25 -0.25 X 1    Y 1    Z 1    W 1    X 2
5   0.25 -0.25 Z 1    Y 1    X 1    X 2    W 1
5   0.25 -0.25 Z 1    Y 1    X 1    W 1    X 2
```

********** THE TIME IS NOW = 4.731 SECONDS ********** CYCLIC:

EQUALS

$+ P_Z^2 P_Y^1 P_X^2)$

$P_Y^1 P_X^2)$

$P_Z^2 P_Y^1)$